# Approximating Commute Times by Truncated Eigenpair Summation


**Brent Smith[1], Van Emden Henson[2], Sarah Powers[2], Geoff Sanders[2]**

[1]Mt. Eden High School, Hayward, CA [2]Lawrence Livermore National Laboratory, Livermore, CA

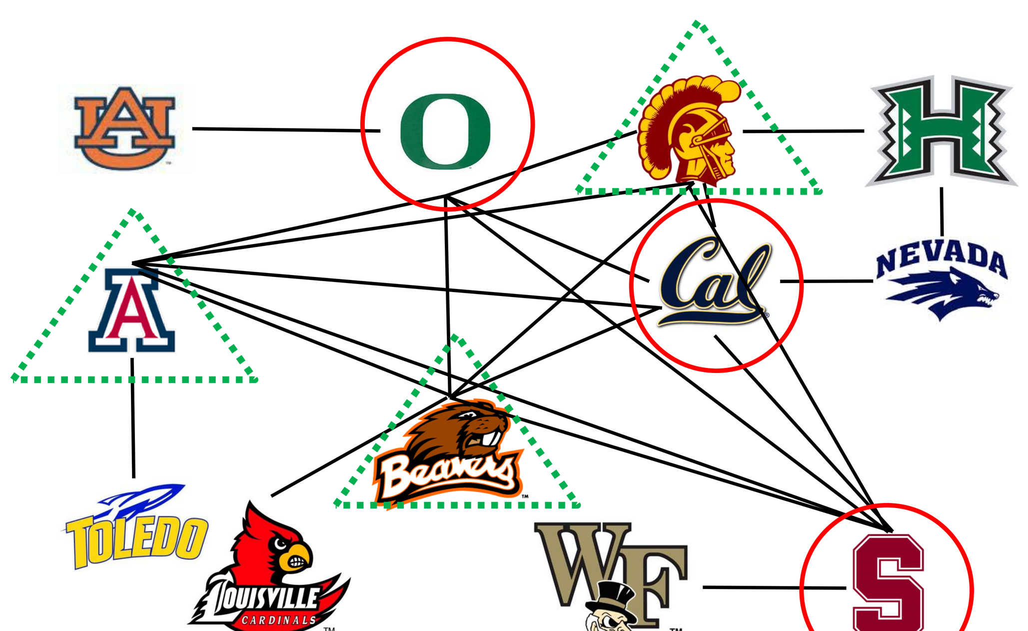


LLNL-POST-491642


## A Model Problem

1

•Suppose I am given a graph where each node is a college football team and each edge is a game played between two teams. Given Cal, Stanford, and Oregon as a "seed set," can I identify the remaining members of the Pac-10, while distinguishing them from teams belonging to other conferences?

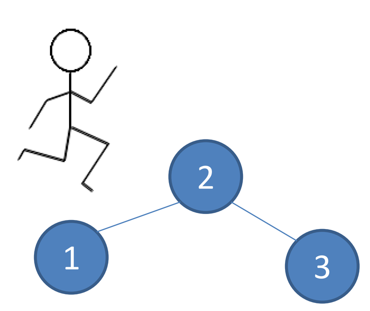


## Introduction: Commute Time Metric

2

•Expected time for a random walker to start at node *i*, arrive at node *j*, and return back to *i*

Commute time between Nodes 1 and 3:
One walker's path: 1->2->1->2->3->2->1 (6 hops)
Another's: 1->2->3->2->3->2->3->2->3->2->1 (10 hops)
Another's: 1->2->3->2->1 (5 hops)….etc
Combining all possibilities: C(1,3) = 8 hops

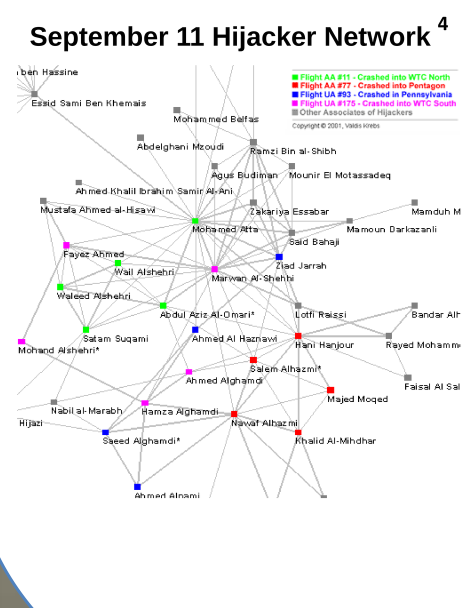


•Measures how "close" two nodes are in a more sophisticated way than simple shortest path, because it considers all possible paths between nodes

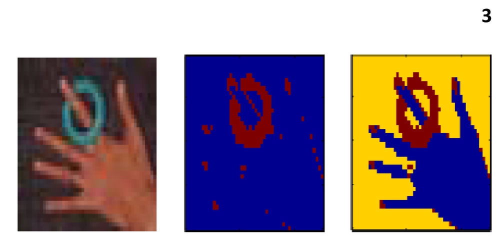
•Robust to changes in graph structure, noise

•Applications include clustering, collaborative filtering (recommendations), and image segmentation

September 11 Hijacker Network [4]



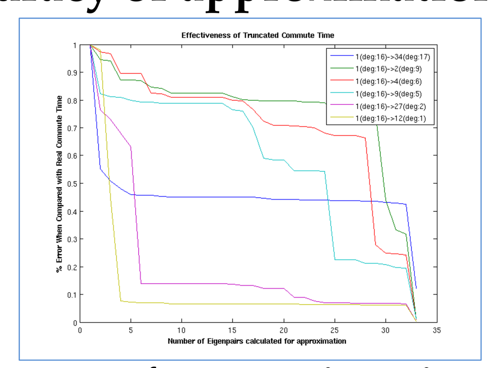

3



## Results

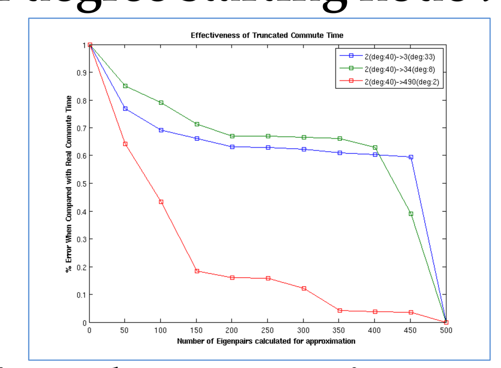
5

• Accuracy of approximations from high degree starting node *i*



34 vertices
78 edges
Zachary's
Karate Club

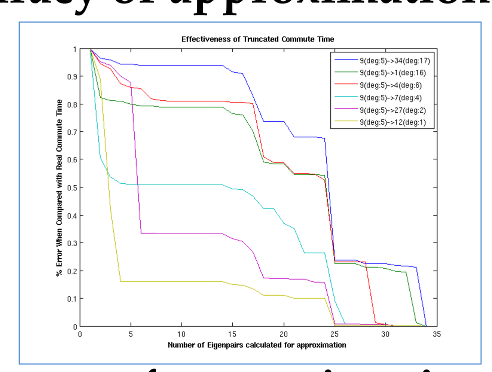
Figure 1



1000 vertices,
~1986 edges
Preferential Attachment
(2 edges added each round)

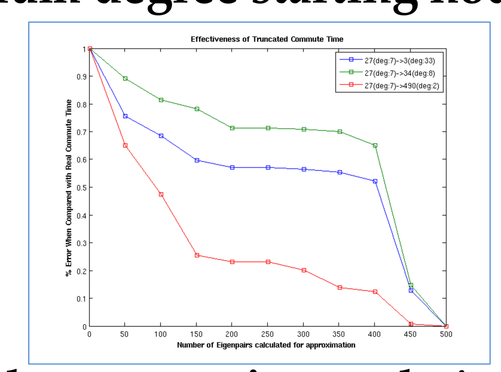
Figure 2

•Accuracy of approximations from medium degree starting node *i*



34 vertices,
78 edges
Zachary's
Karate Club

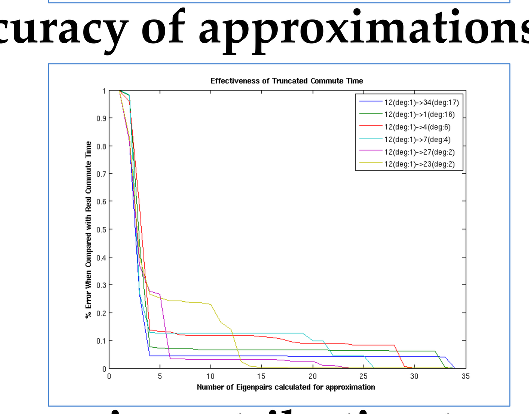
Figure 3



1000 vertices,
~1986 edges
Preferential Attachment
(2 edges added each round)

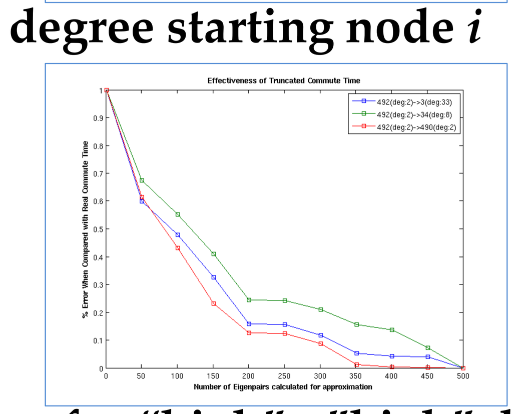
Figure 4

•Accuracy of approximations from low degree starting node *i*



34 vertices,
78 edges
Zachary's
Karate Club

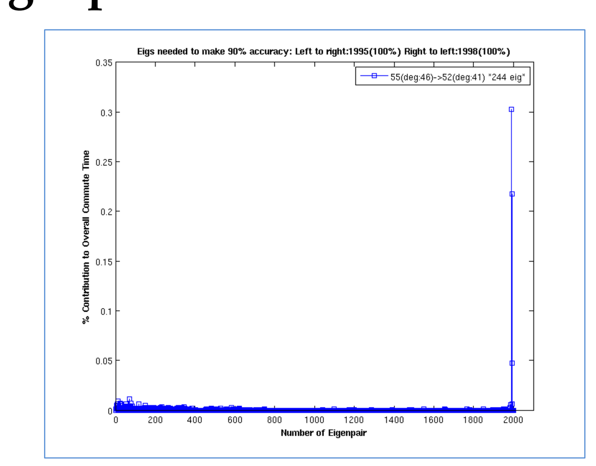
Figure 5



1000 vertices,
~1986 edges
Preferential Attachment
(2 edges added each round)

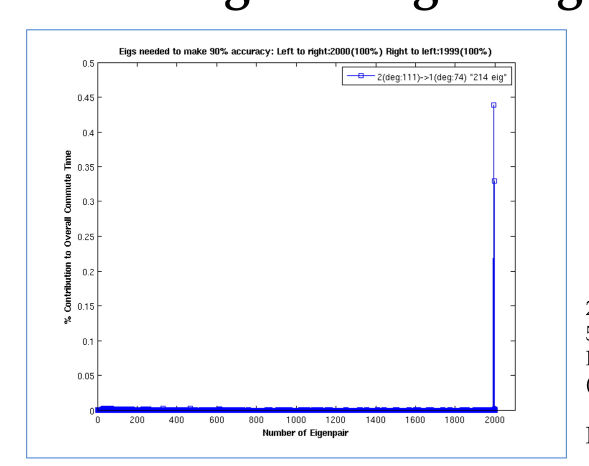
Figure 6

•Eigenpair contribution to commute time for "high"->"high" degree



2000 vertices,
3983 edges
Pref. Attachment
(2 edges added)

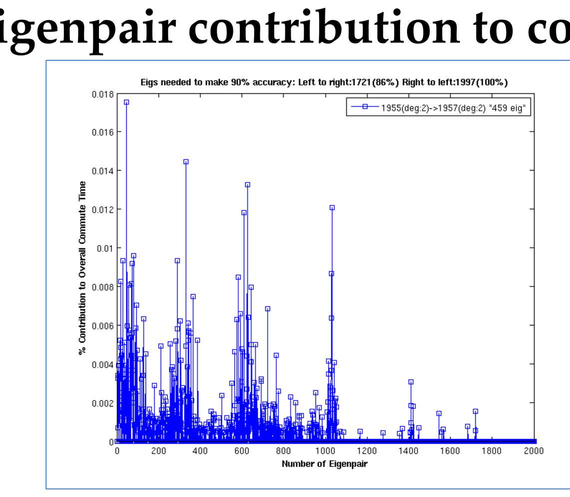
Figure 7



2000 vertices,
5967 edges
Preferential Attachment
(3 edges added each)

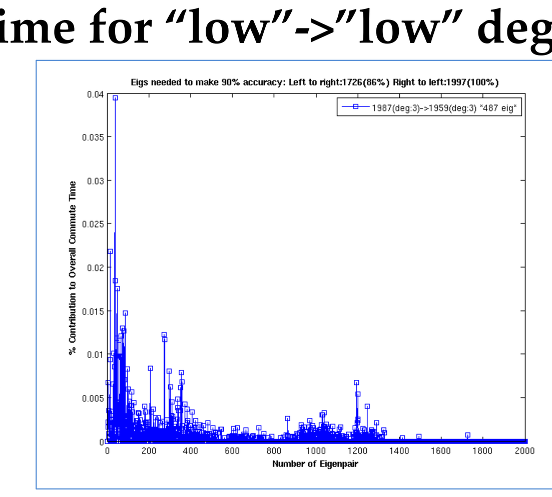
Figure 8

•Eigenpair contribution to commute time for "low"->"low" degree



2000 vertices,
3983 edges
Pref. Attachment
(2 edges added)

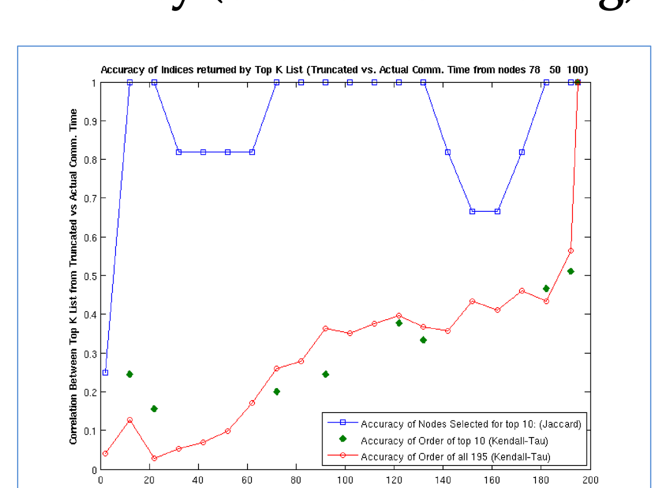
Figure 9



2000 vertices,
5967 edges
Preferential Attachment
(3 edges added each)

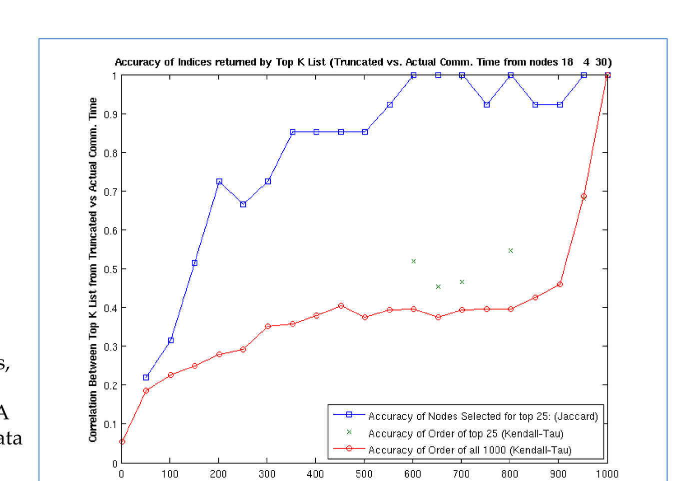
Figure 10

•Accuracy (elements/ordering) of top-k closest commute time nodes




195 vertices,
806 edges
2001 NCAA
Football Data

Figure 11




1000 vertices,
~1986 edges
Preferential
Attachment
(2 edges each)

Figure 12

Jaccard Index , $J(X,Y) = \left|\frac{X \cap Y}{X \cup Y}\right|$  Kendall Tau rank correlation coefficient, $K(X,Y) = \frac{(\#\text{ of concordant pairs}) - (\#\text{ of discordant pairs})}{\frac{1}{2}n(n-1)}$

The pair (a,b) is concordant if the order of a and b are the same on both lists X and Y, and discordant otherwise

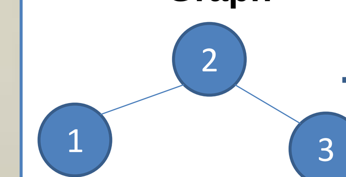
## Calculating Commute Time

3

•Sum up the squared difference of the *i*th and *j*th entries of all eigenvectors of the Graph Laplacian. Each squared difference is weighted by the reciprocal of the corresponding eigenvalue; entire sum is weighted by the volume, $V_G$ (number of edges), of graph

$$C(i,j) = V_G\left[\sum_{k=2}^{n}\frac{1}{\lambda_k}(\phi_k(i)-\phi_k(j))^2\right]$$

**Graph** **Graph Laplacian (L)** **Calculate eigenpairs of L** **Sum for Commute Time**



$$\begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & -.7071 & .4082 \\ 1 & 0 & -.816 \\ 1 & .7071 & .4082 \end{bmatrix}$$

$\lambda_1 = 0, \lambda_2 = 1, \lambda_3 = 3$

Shows connections  Degree  Unused

$C(1,3) = 4[\frac{1}{1}(-.7071-.7071)^2 + \frac{1}{3}(.4082-.4082)^2]$

$C(1,3) = 4[1(-\sqrt{2})^2 + \frac{1}{3}(0)^2]$

$C(1,3) = 4(2) = 8$

•Alternate methods: resistance distance (electron as random walker), Green's function

•PROBLEM: too slow for real-life graphs, which can stretch into millions x millions or billions x billions

## Approach

4

•Tested how closely commute times calculated by truncating the eigen-summation approximate the actual commute time

•Plotted accuracy of commute time approximation as more and more eigenpairs were added (Figures 1-6)

•Checked which specific eigenpairs contributed most to commute time (Figures 7-10)

•Applied truncated commute time to see accuracy (elements chosen, ranking order) of various top-k lists (Figures 11, 12)

## Conclusions

6

•High degree to high degree commute times were best approximated through eigenpairs associated with the largest eigenvalues

•Low degree to low degree commute times were approximated (but not as accurately) by eigenpairs with smallest nonzero eigenvalues

•Attaining accurate Top-k estimations and orderings apparently requires a sizeable fraction of the spectrum. Ordering of the overall list improved more predictably than the ordering of the Top-k list.


This material is based upon work supported by the S.D. Bechtel, Jr. Foundation and by the National Science Foundation under Grant No. 0952013 and Grant No. 0934931. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the S.D. Bechtel, Jr. Foundation or the National Science Foundation.



3 Huaijun Qiu; Hancock, E.R.; , "Clustering and Embedding Using Commute Times," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* , vol.29, no.11, pp.1873-1890, Nov. 2007
4 V.E. Krebs. Uncloaking terrorist networks. First Monday, 7(4), 2002.



This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344